UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/804,320	03/12/2001	Frank S. Caccavale	10830.0075.NPUS00	1069

27927   7590   02/02/2007
RICHARD AUCHTERLONIE
NOVAK DRUCE & QUIGG, LLP
1000 LOUISIANA
53RD FLOOR
HOUSTON, TX 77002

EXAMINER
TRUONG, THANHNGA B

ART UNIT	PAPER NUMBER
2135	

MAIL DATE	DELIVERY MODE
02/02/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

PTOL-90A (Rev. 10/06)

APPLICATION NO./ CONTROL NO.	FILING DATE	FIRST NAMED INVENTOR / PATENT IN REEXAMINATION	ATTORNEY DOCKET NO.
09/804320	3/21/01	Frank S. Caccavale	10830.0075.NPUS00

EXAMINER
Thanhnga B. Truong

ART UNIT	PAPER
2135	013007

DATE MAILED:

Please find below and/or attached an Office communication concerning this application or proceeding.

Commissioner for Patents

This communication letter is sent to applicant per request from BPAI (before the board of patent appeals and interference) mailed on 10/26/06. The attachment is the revised examiner's answer with correction that required from BPAI. Applicant does not need to reply to this action.

Thanhnga (Tanya) B. Truong
AU2135

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

MAILED

FEB 0 2 2007

Application Number: 09/804,320

Filing Date: March 12, 2001

Appellant(s): CACCAVALE, FRANK S.

Technology Center 2100

Richard C. Auchterlonie
For Appellant

Revised
EXAMINER'S ANSWER

This is in response to the appeal brief filed December 09, 2005 appealing from the Office action mailed December 01, 2005.

(1) Real Party in Interest

The statement identifying the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct. The changes are as follows:

Claims 1-10, 13-15, 20-30, 38-39 are pending for rejection. Claims 11, 12 and 31 are objected. Claims 16, 17-19, 32-37, and 40 are allowed.

(4) Status of Amendments After Final

No amendment after final has been filed.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is substantially correct. The changes are as follows:

Claims 1-7, 9, 13-15, 20-26, 28, and 38-39 are rejected under 35 U.S.C. 102(b) as being anticipated by Chen et al (US 5,960,170).

Claim 10 is rejected under 35 U.S.C. 103(a) as being unpatentable over Chen, and further in view of Cassagnol et al (US 6,385,727 B1).

Claims 8, 27, and 29-30 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chen, and further in view of Cassagnol et al (US 6,385,727 B1), and Lam et al (US 5,926,636 A).

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

Chen et al	US 5,960,170	September 28, 1999
Cassagnol et al	US 6,385,727 B1	May 07, 2002
Lam et al	US 5,926,636	July 20, 1999

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claims 1-7, 9, 13-15, 20-26, 28, and 38-39 are rejected under 35 U.S.C. 102(b) as being anticipated by Chen et al (US 5,960,170).

      a.   *Referring to claim 1:*

          i.   Chen teaches:

            (1)   the first file server responding to a request from the client for access to the file in the first file server by determining that an anti-virus scan of the file should be performed, and initiating the anti-virus scan of the file by sending to the second file server a request for the anti-virus scan including a specification of the file **[i.e., viruses are iteratively detected at a client computer. A substantial portion of the tools and information required for the detection and treatment of viruses is provided in a centralized location such as a server, preferably an internet or world wide web server. This virus detection server operates in conjunction with a client to determine whether viruses reside at the client. A virus scan is initiated when a request is received or directed at the virus detection server. The request is direct or can be initiated by various triggering events, such as a programmed request from the client that does not require ongoing user initiation such that the scan is initiated without a request that it apparent to the user (column 2, line 62 through column 3, line 7). In addition, referring to Figure 2, in an initial step 205, a request for a virus scan is received, typically from a source external to the virus detection server 400 such as a client to be scanned. After receipt of the request, in step 210 it is determined by the virus detection server 400 whether a scan is to be performed. Preferably, a validation of the virus scan request is performed pursuant to the determination of whether a scan is to be performed (column 6, lines 34-40)]**, and then

            (2)   the second file server responding to the request for the anti-virus scan by invoking the virus checker program to perform an anti-virus scan of the specified file by obtaining file data of the file from the first file server and storing the file data of the file into the random access memory of the second file server and performing the anti-virus scan upon the file data of the file in the random access

memory [i.e., referring to Figures 4A, the memory 414 is preferably configured to include routines for the iterative detection of viruses. The configurations are described in further detail with reference to the iterative virus detection module 450a of Figure 4B. Referring now to Figure 4B, an embodiment of an iterative virus detection module ("IVDM") 450b in accordance with Chen's invention is shown to include a scanning module 454, a virus pattern module 456, a virus rules module 458, a cleaning module 460, a cleaning pattern module 462, an access managing module 464, and an access data module 466. The iterative virus detection module 450b, and its referenced modules, includes routines for receiving virus detection requests, validating requests, producing virus detection and treatment objects, receiving the results of the execution of the virus detection and treatment objects, and using the results to produce additional virus detection and treatment objects to ultimately detect viruses and treat them. The iterative virus detection module 450b is typically implemented in software, but can also be implemented in hardware or firmware (column 10, lines 52-67)].

> b. _Referring to claim 2:_
>
>> i.   Chen further teaches:
>>
>>> (1)   wherein the first file server determines that the anti-virus scan of the file should be performed when the client requests the first file server to open the file and the first file server finds that the file has not been checked for viruses [i.e., referring to Figure 2 again, in an initial step 205, a request for a virus scan is received, typically from a source external to the virus detection server 400 such as a client to be scanned. After receipt of the request, in step 210 it is determined by the virus detection server 400 whether a scan is to be performed. Preferably, a validation of the virus scan request is performed pursuant to the determination of whether a scan is to be performed. Continuing with a typical example, the request can be provided by the client 300 in the form a request directed to the virus detection server 400, whereupon the virus detection server 400 can validate the request before proceeding with the determination of whether a virus is associated with the client 300. Preferably, request validation is made by

reference to information stored at or accessible to the virus detection server 400 (column 6, lines 34-48; and column 7, lines 4-61 for further details)].

      c.   *Referring to claims 3-6, 21-25, 39:*

            i.   These claims have limitations that is similar to those of claim 2, thus they are rejected with the same rationale applied against claim 2 above.

      d.   *Referring to claim 7:*

            i.   Chen further teaches:

            (1)   wherein the first file server maintains in nonvolatile memory an indication of files that have not been checked for viruses, an indication of files that are in the process of being checked, and an indication of files that have been found to contain viruses **[i.e., an exemplary indexing of virus signatures and the preferred signature scanning technique are now described with reference to Figure 4c-4d. Referring to Figure 4c, an exemplary data table 475 is shown to include columns for platform, virus type, and virus identification. In the exemplary table 475, each row includes information about a particular virus. The information can be used to determine whether a scanning routine corresponding to the particular virus will be implemented. Of course, various scanning routines will correspond to groups of viruses with common characteristics. The data table 475 provides an example of how various virus information is indexed. Various additional or alternative criteria for determining which scanning and treatment routines to use can be provided. Preferably, information such as that provided in the exemplary data table 475 is provided in memory 414 for access by the IVDM 450a in the selection of virus scanning and treatment routines and, more specifically, in the production of virus detection and treatment objects (column 12, line 54 through column 13, line 5)].**

      e.   *Referring to claim 9:*

            i.   Chen further teaches:

            (1)   wherein the second file server receives the request for the anti-virus scan and indirectly invokes the virus checker program by reporting a file access event to an operating system of the second file server, and the operating

system of the file server responds by invoking the virus checker program to perform the anti-virus scan of the file **[i.e., referring to Figure 5, the request can be initiated directly by a client 300 which accesses the virus detection server 400 using conventional network communication protocols. Although the triggering event 502 that prompts the request 505 is typically initiated directly by the user of the client 300, the request can alternatively be initiated by a triggering event other than user prompting or initiation. This allows for regular virus scanning without requiring user input. Additionally, a group of computers that a user might seek to manage, such as a plurality of computers residing on a LAN, can be subjected to regular virus scanning without requiring user initiation and with minimal use of network resources (column 16, lines 6-17)].**

      f.   *Referring to claim 13:*

      i.   This claim has limitations that is similar to those of claim 1, thus it is rejected with the same rationale applied against claim 1 above.

      g.   *Referring to claims 14, 18:*

      i.   These claims have limitations that is similar to those of claims 2 and 3, thus they are rejected with the same rationale applied against claims 2 and 3 above.

      h.   *Referring to claims 15, 28:*

      i.   These claims have limitations that is similar to those of claim 9, thus they are rejected with the same rationale applied against claim 9 above.

      i.   *Referring to claim 20:*

      i.   Chen teaches:

      (1)   at least one client; a first file server coupled to the client for access of the client to at least one file in the first file server **[i.e., referring to Figure 7, an exemplary network communication system 700 includes a local area network (LAN) with clients 300c, a gateway server 710 and an administrative server 750 (column 24, lines 21-24)];** and

      (2)   at least a second file server coupled to the first file server for data access of the second file server to the file in the first file server, the

second file server being programmed with a virus checker program, the virus checker program being executable by the second file server to perform an anti-virus scan upon file data in random access memory of the second file server; wherein the first file server is programmed to respond to a request from the client for access to the file in the first file server by determining that an anti-virus scan of the file should be performed, and initiating the anti-virus scan of the file by sending to the second file server a request for the anti-virus scan including a specification of the file; and the second file server is programmed to respond to the request for the anti-virus scan by invoking the virus checker program to perform an anti-virus scan of the specified file by obtaining file data of the file from the first file server and storing the file data of the file in the random access memory of the second file server and performing the anti-virus scan upon the file data in the random access memory **[i.e., these limitations are similar to those of claim 1, thus they are rejected with the same rationale applied against claim 1 above].**

j.   *Referring to claim 26:*

i.   This claim has limitations that is similar to those of claim 7, thus it is rejected with the same rationale applied against claim 7 above.

k.   *Referring to claim 38:*

i.   This claim has limitations that is similar to those of claim 20, thus it is rejected with the same rationale applied against claim 20 above.

Claim 10 is rejected under 35 U.S.C. 103(a) as being unpatentable over Chen, and further in view of Cassagnol et al (US 6,385,727 B1).

a.   *Referring to claim 10:*

i.   Chen teaches the claimed subject matter, however Chen does not explicitly mention:

(1)   processes executing in a user mode and processes executing in a kernel mode

ii.   However, Cassagnol teaches:

(1)   In some embodiments, the first processor has a kernel mode of operation and a user mode of operation, and the kernel mode and the

user mode define separate security cells. In such embodiments, the first processor preferably executes non-secure software in the user mode of operation and secure software in the kernel mode of operation **(column 3, lines 25-30).**

       iii.   It would have been obvious to a person having ordinary skill in the art at the time the invention was made to:

       (1)   mention or include the two security cells, user mode and kernel mode (in Chen's CPU 802 of Figure 8A), since the CPU 802 performs functions under the guidance and control provided by instructions received from memory 804, the functions including communications through network media 812 using the network interface 810 **(column 24, lines 46-50 of Chen).**

       iv.   The ordinary skilled person would have been motivated to:

       (1)   mention or include the two security cells, user mode and kernel mode (in Chen's CPU 802 of Figure 8A), because Computer viruses continue to be problematic to computers and computer users. Such viruses are typically found within computer programs, files, or code and can produce unintended and sometimes damaging results **(column 1, lines 13-15 of Chen).**

Claims 8, 27, 29-30 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chen, and further in view of Cassagnol et al (US 6,385,727 B1), and Lam et al (US 5,926,636).

       a.   *Referring to claim 8:*

       i.   Chen teaches the claimed subject matter except for:

       (1)   wherein the request for the anti-virus scan including a specification of the file is an Open Network Computing Remote Procedure Call.

       ii.   However, Lam teaches:

       (1)   Communications between a client and a server over heterogeneous network 100 require a method for transporting requests over network 100 from a client running under one operating system to a server that is either running under another operating system, or the same operating system. One widely used method for communication over heterogeneous network 100 is a remote procedure call (RPC). Techniques for implementing client/server applications, and client/server

applications with remote procedure calls are known to those skilled in the art. A remote procedure call (RPC) hides the physical structure of network 100 and makes a server on network 100 appear to be one function call away. Specifically, a remote procedure call hides the details of network 100 by using a procedure call mechanism that is well known **(column 1, lines 44-58).**

        iii.   It would have been obvious to a person having ordinary skill in the art at the time the invention was made to:

        (1)   include the remote procedure call (in Chen's invention) for performing operations over a network, such as network 100, is a client/server architecture **(column 1, lines 28-30 of Lam).**

        iv.   The ordinary skilled person would have been motivated to:

        (1)   allow a client to interoperate with one or more servers on other computing platforms, even when the client and server are from different vendors with different operating systems **(column 1, lines 64-66 of Lam).**

        b.   *Referring to claim 27:*

        i.   This claim has limitations that is similar to those of claim 8, thus it is rejected with the same rationale applied against claim 8 above.

        c.   *Referring to claim 29:*

        i.   This claim has limitations that is similar to those of claim 10, thus it is rejected with the same rationale applied against claim 10 above.

        d.   *Referring to claim 30:*

        i.   This claim has limitations that is similar to those of claim 11, thus it is rejected with the same rationale applied against claim 11 above.

(10a) Allowable Subject Matter

Claims 11, 12, 30, and 31 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

Claims 16, 17-19, 32-37, 40 are allowed.

(10b) Response to Arguments

Appellant's arguments filed December 09, 2005 have been fully considered, however, claims 1-10, 13-15, 20-29, 38-39 are still not persuasive.

Regarding to the Appellant's arguments on anticipation by Chen. Appellant states that "the first file server responding to a request from the client for access to the file in the first file server by determining that an anti-virus scan of the file should be performed, and initiating the anti-virus scan of the file by sending to the second file server a request for the anti-virus scan including a specification of the file; the second file server responding to the request for the anti-virus scan by invoking the virus checker program to perform an anti-virus scan of the specified file by obtaining file data of the file from the first file server and storing the file data of the file into the random access memory of the second file server and performing the anti-virus scan upon the file data of the file in the random access." This is not done in Chen et al.

Chen does teach the claimed subject matter. In fact, Chen fully and clearly discloses a virus detection server is provided for centralized access and iterative techniques are provided to effectively detect and treat viruses. In one aspect of the Chen's invention, viruses are iteratively detected at a client computer. A substantial portion of the tools and information required for the detection and treatment of viruses is provided in a centralized location such as a server, preferably an internet or world wide web server. This virus detection server operates in conjunction with a client to determine whether viruses reside at the client. A virus scan is initiated when a request is received or directed at the virus detection server. The request is direct or can be initiated by various triggering events, such as a programmed request from the client that does not require ongoing user initiation such that the scan is initiated without a request that it apparent to the user. Once it is determined by the virus detection server that a valid virus detection request has been received, the virus detection server operates to iteratively detect and treat viruses associated with the requester, typically the client. The iterative production of virus detection objects allows objects to be specifically tailored according to previously determined conditions and/or conditions discovered as a result of the execution of previously produced virus detection objects. Specifically, a virus detection object is produced by the virus detection server and is transmitted to the

client. The virus detection object includes an executable program which the client includes a corresponding executing engine. Thus, when the client receives the virus detection object, it executes the object and produces a result that is transmitted back to the virus detection server. The results of the execution of the virus detection object are transmitted to the virus detection server so that the server can produce additional virus detection objects based upon the results of the execution of the previous virus detection object or objects (column 2, lines 59-67 through column 3, lines 1-27 of Chen). There are various methods for detecting viruses. One method of detection is to compare known virus signatures to targeted files to determine whether the targeted files include a virus signature and, thus, the corresponding virus. The comparison data used for virus detection might include a set of such known virus signatures and, possibly, additional data for virus detection. Typically, the comparison data is maintained in a computer storage medium for access and use in the detection of viruses. For example, for a personal computer the comparison data might be stored on the computer's hard disk (column 1, lines 34-44 of Chen). Furthermore, Chen teaches in Figure 4A, a virus detection server 400 in accordance with the present invention is shown. The virus detection server 400 includes a CPU 412, memory 414, a data storage device 416 such as a hard disk, I/O ports 418 and a network interface 420. The CPU 412 is conventional such as a Pentium Pro as provided by Intel Corporation, Santa Clara, Calif. The memory 414 is preferably conventional RAM but may also include conventional ROM. Additionally, the memory 414 is preferably configured to include routines for the iterative detection of viruses.Thus, Chen teaches the claimed subject matter of claim 1.

It is therefore shown that the components disclosed by Chen constitute the claimed first file server responding to a request from the client for access to the file in the first file server by determining that an anti-virus scan of the file should be performed, and initiating the anti-virus scan of the file by sending to the second file server a request for the anti-virus scan including a specification of the file; the second file server responding to the request for the anti-virus scan by invoking the virus checker program to perform an anti-virus scan of the specified file by obtaining file data of the file from the first file server and storing the file data of the file into the random access memory of the

second file server and performing the anti-virus scan upon the file data of the file in the random access. Chen teaches in Figure 1, **the interconnections between a client and a virus detection server.** The virus detection server and the method of virus detection of Chen's invention preferably operate on **a system 100 wherein a client-server relationship can be established between the virus detection server 400 and any one of a plurality of clients 300.** In the system 100 illustrated in Figure 1, the virus detection server 400 resides on a wide area network (WAN) such as the network typically referred to as the Internet or World Wide Web. **Various exemplary interconnections are shown between clients 300 and a virus detection server 400. In one example, the client 300a is shown coupled via line 10 to the WAN to allow communication between the client 300a and the virus detection server 400.** Storing data in also met in Chen's Figure 4A using memory 414 as RAM or ROM.

Claim 20 consists a system claim, which met on column 28, lines 36-53 of Chen, to implement claim 1 and is rejected by the same reasons as given above for claim 1.

Regarding Appellant's arguments to claim 2 and 21 that Chen fails to disclose that the first file server determines that the anti-virus scan of the file should be performed when the client requests the first file server to open the file and the first file server finds that the file has not been checked for viruses.

Examiner does not agree with the appellant and still maintain that Chen teaches the claimed subject matter. Referring to Figure 2 again, in an initial step 205, a request for a virus scan is received, typically from a source external to the virus detection server 400 such as a client to be scanned. **After receipt of the request, in step 210 it is determined by the virus detection server 400 whether a scan is to be performed. Preferably, a validation of the virus scan request is performed pursuant to the determination of whether a scan is to be performed.** Continuing with a typical example, the request can be provided by the client 300 in the form a request directed to the virus detection server 400, whereupon the virus detection server 400 can validate the request before proceeding with the determination of whether a virus is associated with the client 300. Preferably, request validation is made by reference to information stored at or accessible to the virus detection server 400 (column 6, lines 34-48; and

column 7, lines 4-61 for further details). In addition, there are various methods for detecting viruses. One method of detection is to compare known virus signatures to targeted files to determine whether the targeted files has been checked to include a virus signature and, thus, the corresponding virus (column 1, lines 34-37 of Chen). Thus, Chen teaches the claimed subject matter of claims 2 and 21.

It is therefore shown that the components disclosed by Chen constitute the claimed the first file server determines that the anti-virus scan of the file should be performed when the client requests the first file server to open the file and the first file server finds that the file has not been checked for viruses.

Regarding Appellant's arguments to claims 3 and 22 that Chen fails to disclose that the first file server determines that the anti-virus scan of the file should be performed when the client requests a file to be closed after the client writes to the file. These claims 3 and 22 consist a inversion of claim 2 and 21. Therefore, they are rejected by the same reasons as given above for claims 2 and 21.

Claims 5, 24, and 39 consists similar limitations that discloses in claims 2 and 3 and is rejected by the same reasons as given above for claims 2 and 3.

Regarding Appellant's arguments to claims 6 and 25 that Chen fails to disclose that the first file server determines that an additional anti-virus scan of the file should not be performed in response to the access of the file by the virus checker program.

Examiner does not agree with the appellant and still maintain that Chen teaches the claimed subject matter. Referring to Figure 2 again, in an initial step 205, a request for a virus scan is received, typically from a source external to the virus detection server 400 such as a client to be scanned. **After receipt of the request, in step 210 it is determined by the virus detection server 400 whether a scan is to be performed or not. Preferably, a validation of the virus scan request is performed pursuant to the determination of whether a scan is to be performed or not.** Continuing with a typical example, the request can be provided by the client 300 in the form a request directed to the virus detection server 400, whereupon the virus detection server 400 can validate the request before proceeding with the determination of whether a virus is associated with the client 300. Preferably, request validation is made by reference to

information stored at or accessible to the virus detection server 400 (column 6, lines 34-48;. and column 7, lines 4-61 for further details).   Thus, Chen teaches the claimed subject matter of claims 6 and 25.

It is therefore shown that the components disclosed by Chen constitute the claimed the first file server determines that an additional anti-virus scan of the file should not be performed in response to the access of the file by the virus checker program.

Regarding Appellant's arguments to claim 7 that Chen fails to disclose wherein the first file server maintains in nonvolatile memory an indication of files that have not been checked for viruses, an indication of files that are in the process of being checked, and an indication of files that have been found to contain viruses.

Examiner does not agree with the appellant and still maintain that Chen teaches the claimed subject matter.   In fact, Chen discloses in the exemplary table 475, each row includes information about a particular virus.   The information can be used to determine whether a scanning routine corresponding to the particular virus will be implemented. Of course, various scanning routines will correspond to groups of viruses with common characteristics.   The data table 475 provides an example of how various virus information is indexed.   Various additional or alternative criteria for determining which scanning and treatment routines to use can be provided.   **Preferably, information such as that provided in the exemplary data table 475 is provided in memory 414, wherein the memory 414 is preferably conventional RAM but may also include conventional ROM, which is non-volatile memory (column 10, lines 25-26 of Chen),** for access by the IVDM 450a in the selection of virus scanning and treatment routines and, more specifically, in the production of virus detection and treatment objects (column 12, line 54 through column 13, line 5). Appellant claimed language is nothing more than just storing different type of files or data in the memory. Thus, Chen teaches the claimed subject matter of claim 7.

It is therefore shown that the components disclosed by Chen constitute the claimed the first file server maintains in nonvolatile memory an indication of files that

have not been checked for viruses, an indication of files that are in the process of being checked, and an indication of files that have been found to contain viruses.

Regarding Appellant's arguments to claim 9 that Chen fails to disclose that a triggering event received by the file server is directed inside the file server to invoke a virus checker program to perform the anti-virus scan of the file (see page 13 of appellant's appeal brief).

Examiner does not agree with the appellant and still maintain that Chen teaches the claimed subject matter. In fact whether a triggering event received by the file server is directed inside or outside the file server, this limitation is not cited in the claimed language. Referring to Figure 5 of Chen, the request can be initiated directly by a client 300 which accesses the virus detection server 400 using conventional network communication protocols. Although the triggering event 502 that prompts the request 505 is typically initiated directly by the user of the client 300, the request can alternatively be initiated by a triggering event other than user prompting or initiation. This allows for regular virus scanning without requiring user input. Additionally, a group of computers that a user might seek to manage, such as a plurality of computers residing on a LAN, can be subjected to regular virus scanning without requiring user initiation and with minimal use of network resources (column 16, lines 6-17). Thus, Chen teaches the claimed subject matter of claim 9.

It is therefore shown that the components disclosed by Chen constitute the claimed a triggering event received by the file server is directed inside the file server to invoke a virus checker program to perform the anti-virus scan of the file.

Claims 13 and 38 consist similar limitations that discloses in claims 1, 2, and 3 and are rejected by the same reasons as given above for claims 1, 2, and 3.

Claims 14 and 18 consist similar limitations that discloses in claims 2 and 3 and are rejected by the same reasons as given above for claims 2 and 3.

Claim 26 consists similar limitations that discloses in claim 7 and is rejected by the same reasons as given above for claim 7.

Regarding Appellant's arguments to claim 10 that it is not seen how the proposed combination of Chen et al. and Cassagnol et al. would provide the appellant's server for

virus checking executing in the user mode that receives the request for the anti-virus scan from the first file server and forwards the request to a virus checker initiator driver executing in the kernel mode, and the virus checker initiator driver executing in the kernel mode initiates a file access event, and the virus checker program initiates the anti-virus scan of the file in response to the virus checker initiator driver initiating the file access event (see page 16 of appellant's appeal brief).

Examiner again disagrees and still maintains that:
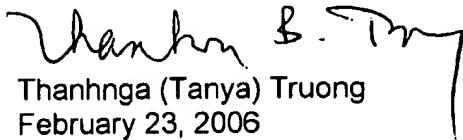
In response to appellant's argument that there is no suggestion to combine the references, the examiner recognizes that obviousness can only be established by combining or modifying the teachings of the prior art to produce the claimed invention where there is some teaching, suggestion, or motivation to do so found either in the references themselves or in the knowledge generally available to one of ordinary skill in the art. See *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988)and *In re Jones*, 958 F.2d 347, 21 USPQ2d 1941 (Fed. Cir. 1992). In this case, the combination of the teachings between Chen and Cassagnol is sufficient. Chen teaches the virus detection server 400 uses a conventional operating system such as UNIX or Windows NT and implements conventional internet communication protocols such as the transmission control protocol/internet protocol (TCP/IP) suite (column 5, lines 66-67 through column 6, lines 1-3 of Chen). Furthermore, the memory 314 is shown to include an operating system 328, a browser 330, and a virus detection shell 332. The operating system 328 is preferably a conventional one for a personal computer such as Windows95 or WindowsNT as provided by Microsoft, Inc. of Redmond, Wash (column 9, lines 12-15 of Chen). It is so obvious that the operation of user mode and kernel mode is part of Chen's operation system. However, Chen is silent about the operation of user mode and kernel mode. Cassagnol, on the other hand, discloses the first processor has a kernel mode of operation and a user mode of operation, and the kernel mode and the user mode define separate security cells. In such embodiments, the first processor preferably executes non-secure software in the user mode of operation and secure software in the kernel mode of operation (column 3, line 25-30 of Cassagnol).

Claim 29 consists similar limitations that discloses in claim 10 and is rejected by the same reasons as given above for claim 10.
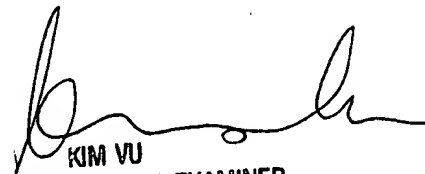
(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

Respectfully submitted,

Thanhnga (Tanya) Truong
February 23, 2006

KIM VU
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100

Conferees
Kim Vu
Hosuk Song

NOVAK DRUCE & QUIGG, LLP
1000 Louisiana, Suite 5320
Houston, TX 77002